|  |  |
| --- | --- |
| Usability Engineering | CS2511 Practical Sessions |

Dynamic Web Interfaces (v)

The aim of this exercise is to combine the code you created in previous exercises into a single web-page that does the following:

* Reads an array, creates an object to represent each element in the array, and sets the position, colour, etc., of the objects according to certain values held in the array.
* Allows the objects to be added or removed from the page depending upon the setting(s) of one or more sliders.

In addition, the page should also provide more detailed information on an object when the mouse is passed over it.

You may find this exercise easier if you tackle it in stages.

Assuming you have successfully completed the previous exercise, you will have a page that creates and positions objects according to data held in an array. Make a copy of this code and use this as the basis of the new page.

Add-in the HTML code for the slider that you created in previous exercises, and adjust the style position settings so that the slider appears at a suitable position on the page.

Add-in the JavaScript code for the slider, and modify it so that it correctly moves the slider-bar along its track.

Next, add-in the code to add or remove objects that you created in previous exercises, then modify the code to check that these objects exist and add or remove them as necessary.

The next stage is to modify the code so that it displays additional information when the mouse is positioned over an object.

First, add a textarea to your page, and position it appropriately.

Next, modify your code so that, each time an object is created, an onmouseover event-handler is attached to it, e.g.:

newObj.onmouseover = showData;

You will also need to 'hide' the data when the mouse is moved off an object, so add an onmouseout event-handler, e.g.:

newObj.onmouseout = hideData;

Now create the necessary functions. The 'showData' function should identify which object the mouse is over and then find its details in the array. This can done by checking the 'id' of the object which generated the onmouseover event, using evt.target.id for standards-compliant browsers such as Firefox and evt.srcElement.id for Internet Explorer.

Once you have obtained the 'id' of the relevant object, you can look it up in the array, format the data into a message, and display it in the textarea.

The 'hideData' function should put an empty string into the textarea, thus obliterating any text already there.